Calculation of Clustered Eigenvalues of Large Matrices Using Variance Minimization Method

EMILI BESALÚ,[1] JOSEP MARIA BOFILL[2]

[1]*Institut de Química Computacional i Departament de Química, Universitat de Girona, Campus de Montilivi, E-17071 Girona, Catalunya, Spain*

[2]*Departament de Química Orgànica, Universitat de Barcelona, Martí i Franquès 1, E-08028 Barcelona, Catalunya, Spain*

Received 7 August 1997; accepted 15 July 1998

ABSTRACT: The autoadjusting perturbation theory method is presented and developed to calculate eigenpairs of a square matrix. The procedures to simultaneously compute a cluster of eigenpairs by variance minimization are also given. Finally, numerical examples are reported. © 1998 John Wiley & Sons, Inc. J Comput Chem 19: 1777–1785, 1998

Keywords: clustered eigenvalues; large matrices; variance minimization method

Introduction

The solution of the eigenvalue equation

$$\mathbf{HC} = \mathbf{CE} \tag{1}$$

of a real symmetric matrix **H** is one of the main problems in quantum chemistry. The numerical representation of the operators associated with the important observables for quantum chemists leads to matrices of very large dimension that generally cannot be stored in the fast memory. Because of this fact, the solution of eq. (1) by iterative methods that require only matrix-vector products has been developed. This is the computational basis of correlated wave functions calculations such as the direct configuration interaction (CI) algorithm.[1]

The iterative methods are all based on the Ritz–Galerkin algorithm[2] for solving equations in a subspace. In this approach, one solves eq. (1) in a subspace using normal diagonalizers. The subspace is generated in an iterative way. The robustness of these methods relies on both the stability of the normal diagonalizer and the efficiency of the iterative construction of the subspace. The iterative

Correspondence to: J. M. Bofill; e-mail: jmbofill@canigo.qo.ub.es

Contract/grant sponsor: Spanish DGICYT; contract/grant number: PB95-0278-C02-01

Contract/grant sponsor: Spanish MEC; contract/grant number: CICYT SAF96-0158

CCC 0192-8651 / 98 / 151777-09

methods can be divided into two groups, depending on the manipulation of the iterative subspace. The first group consists of all the iterative methods such that the subspace is increased at each iteration. In this group falls the well-known Lanczos scheme.[3] In the other group the dimension of the subspace remains constant, but at each iteration all the basis vectors are changed. This type of iterative procedure was initially proposed by Karush.[4]

Almost all the diagonalization algorithms used in quantum chemistry are based on modifications of the Lanczos scheme. The Lanczos method is completely correct but presents some numerical instabilities such as the loss of the eigenvectors' orthogonality. Because of this fact, at each iteration one uses the Gram–Schmidt procedure to orthogonalize the new vectors set with respect to the vectors defining the current subspace. The new orthogonalized vectors together with the oldest vectors will define the basis of the increased subspace. This alternative scheme to the Lanczos method is the so-called Paige variant of the symmetric Lanczos algorithm.[5,6]

One of the most efficient algorithms to diagonalize large matrices in quantum chemistry is that proposed by Davidson.[7,8] Briefly, the Davidson algorithm can be seen as a Paige variant of the symmetric Lanczos algorithm plus a preconditioning derived from first-order perturbation theory to generate the new vectors that become members of the increased subspace basis. Recently, Saad[9] pointed out that the Davidson formula does not lead to an optimal choice to generate the new vectors. Because of this feature, Olsen et al.[10] and Bofill and Anglada[11] proposed an alternative formula that can be seen as an inexact Newton-like formula.

From the authors' knowledge, the algorithm of van Lenthe and Pulay[12] and its improved version of Bofill and Anglada[11] is the sole algorithm in quantum chemistry that falls in the second group.[13] Either within the van Lenthe and Pulay's method[12] or its improved version,[11] the dimension of the subspace is equal to 3. The use of a subspace of dimension 3 to solve eq. (1) can be justified[11,12] from the theory of the conjugate gradient methods.[14] We note that the performance of this method is quite competitive with respect to the Davidson method.[11] Recently, Davidson[7] proposed a combination of van Lenthe and Pulay's method[12] with a variance minimization.

The above two groups of iterative methods can be easily extended to solve the generalized symmetric eigenvalue problem,[13]

$$\mathbf{HC} = \mathbf{SCE}, \tag{2}$$

where **S** is a symmetric and positive definite matrix. Some methods to solve eq. (2) are based on the inversion of the **S** matrix and after this the Davidson algorithm is applied.[15] However, this procedure is impractical for very large matrices. An elegant way was proposed in refs. 13 and 16, which only involves the **H** and **S** matrices multiplication on a vector.

In this article we present a method to diagonalize large matrices based on variance minimization. This technique was introduced some time ago by Feler,[17] but it is coupled with the so-called method of optimal relaxation (MOR) of Shavitt et al.[18] More recently, Wood and Zunger[19] coupled this variance minimization with the Davidson method as the way to generate basis vectors for building the iterative subspace. These methods were called residual minimization-direct inversion in the iterative subspace (RMM-DIIS) because of the strong relation in this context of the variance minimization with Pulay's convergence acceleration technique.[20] However, Wood and Zunger did not present any study focusing on the truncation of the iterative subspace and did not indicate how the method should be employed to find a cluster of eigenpairs. Finally, in the present study, the iterative subspace is generated using the vectors resulting from the autoadjusting perturbation theory (APT).[21]

The article is divided into three parts: first, we comment on the APT theory;[21] second, the variance minimization technique applied in the subspace of the APT vectors is presented, and third, some examples are shown.

Theoretical Basis

GENERATION OF ITERATIVE SUBSPACE

One of the important questions in solving the eigenvalue equation by the approximation from a subspace is how to generate the basis vectors of this subspace. The quality of this basis is crucial to achieving good convergence. To this aim we propose the APT of Besalú and Carbó–Dorca[21] as the way to generate the basis vectors. This technique is based on the iterative use of similarity transforma-

tions to diagonalize the **H** matrix, that is,

$$\cdots \left[(\mathbf{Z}^{(i)})^{-1} \cdots \left[(\mathbf{Z}^{(2)})^{-1} \left[(\mathbf{Z}^{(1)})^{-1} \mathbf{H} \mathbf{Z}^{(1)} \right] \mathbf{Z}^{(2)} \right] \cdots \mathbf{Z}^{(i)} \right] \cdots \rightarrow \mathbf{E}, \quad (3)$$

where $\{\mathbf{Z}^{(i)}\}_{i=1}^{\text{iterations}}$ are the similarity transformation matrices applied along the iterative process. During the successive similarity transformations on the **H** matrix, the resulting matrices tend to be a matrix **E**. At the convergence, a portion of this matrix **E** is diagonal and contains all desired eigenvalues of the **H** matrix. The APT method is based on the particular structure of the $\mathbf{Z}^{(i)}$ matrices when one is interested in a particular eigenvector. In this situation each $\mathbf{Z}^{(i)}$ matrix may be written as

$$\mathbf{Z}^{(i)} = \mathbf{I} + \mathbf{W}^{(i)} \quad \forall i, \quad (4)$$

where **I** is the unit matrix and $\mathbf{W}^{(i)}$ is a matrix filled with zeros except a column, let us say column p, denoted $\mathbf{w}_p^{(i)}$. Also, $\mathbf{W}_{pp}^{(i)} = (\mathbf{w}_p^{(i)})_p = 0$, where $(\mathbf{w}_p^{(i)})_r$ stands for the rth element of the vector $\mathbf{w}_p^{(i)}$. In this way, equating each column of each side of eq. (4), we find the relationship

$$\begin{cases} \mathbf{z}_j^{(i)} = \mathbf{e}_j & j \neq p, \\ \mathbf{z}_p^{(i)} = \mathbf{e}_p + \mathbf{w}_p^{(i)} \end{cases} \quad (5)$$

where $\{\mathbf{z}_j^{(i)}\}$ are the column vectors of the $\mathbf{Z}^{(i)}$ matrix and $\{\mathbf{e}_j\}$ are the unit vectors [e.g., $\mathbf{e}_p^T = (0, \ldots, 1, \ldots, 0)$], where the number 1 is the pth position. The inverse of any $\mathbf{Z}^{(i)}$ matrix is

$$(\mathbf{Z}^{(i)})^{-1} = \mathbf{I} - \mathbf{W}^{(i)}. \quad (6)$$

We note that a set of $\mathbf{Z}^{(i)}$ matrices has the same structure if each one of the respective $\mathbf{W}^{(i)}$ matrices bears the nonzero column vector in the same position. With this definition we observe that a product of either $\mathbf{Z}^{(i)}$ matrices with the same structure or its inverses gives another $\mathbf{Z}^{(i)}$ matrix or its inverse with the same structure, respectively. That is,

$$\prod_{i=1}^{n} \mathbf{Z}^{(i)} = \mathbf{I} + \sum_{i=1}^{n} \mathbf{W}^{(i)} = \mathbf{I} + \mathbf{W}^{[n]} = \mathbf{Z}^{[n]}, \quad (7)$$

where the implicit definitions taken into account are evident. All the $\{\mathbf{W}^{(i)}\}$ matrices and the $\mathbf{W}^{[n]}$ matrix have the nonzero column vector in the same position, let us say column p. Finally, we note that from a geometrical point of view the elements of the $\mathbf{w}_p^{(i)}$ vector correspond to a set of "polar angles."

From a strictly computational viewpoint, one only needs to store the $\mathbf{w}_p^{(i)}$ vector. The product of $\mathbf{H}\mathbf{Z}^{(i)}$ only consists of adding the vector $\mathbf{H}\mathbf{w}_p^{(i)}$ to the pth column of the **H** matrix. When the similarity transformation is applied to the **H** matrix, the resulting transformed matrix $\mathbf{H}' \leftarrow [(\mathbf{Z}^{(i)})^{-1} \mathbf{H} \mathbf{Z}^{(i)}]$ possesses the following structure:

$$\begin{aligned} \mathbf{H}' \leftarrow \mathbf{H} &- \left(\mathbf{w}_p^{(i)} \mathbf{H}_{p1} | \cdots | \mathbf{w}_p^{(i)} \mathbf{H}_{pp} | \cdots | \mathbf{w}_p^{(i)} \mathbf{H}_{pn} \right) \\ &+ \left(\mathbf{0} | \cdots | \mathbf{H}\mathbf{w}_p^{(i)} | \cdots | \mathbf{0} \right) \\ &- \left(\mathbf{0} | \cdots | \mathbf{w}_p^{(i)} \left(\mathbf{H}\mathbf{w}_p^{(i)} \right)_p | \cdots | \mathbf{0} \right). \end{aligned} \quad (8)$$

Equation (8) can be decomposed by elements of the $\mathbf{H}'$ matrix, $\{\mathbf{H}'_{rs}\}$, obtaining the next four cases:

$$r = p, \quad s = p: \quad \mathbf{H}'_{pp} \leftarrow \mathbf{H}_{pp} + \left(\mathbf{H}\mathbf{w}_p^{(i)} \right)_p; \quad (9a)$$

$$r \neq p, \quad s = p: \quad \mathbf{H}'_{rp} \leftarrow \mathbf{H}_{rp} - \left(\mathbf{w}_p^{(i)} \right)_r \mathbf{H}_{pp} + \left(\mathbf{H}\mathbf{w}_p^{(i)} \right)_r - \left(\mathbf{w}_p^{(i)} \right)_r \left(\mathbf{H}\mathbf{w}_p^{(i)} \right)_p; \quad (9b)$$

$$r = p, \quad s \neq p: \quad \mathbf{H}'_{ps} \leftarrow \mathbf{H}_{ps}; \quad (9c)$$

$$r \neq p, \quad s \neq p: \quad \mathbf{H}'_{rs} \leftarrow \mathbf{H}_{rs} - \left(\mathbf{w}_p^{(i)} \right)_r \mathbf{H}_{ps}. \quad (9d)$$

We recall that $\mathbf{H}\mathbf{w}_p^{(i)}$ and $\mathbf{w}_p^{(i)}$ are column vectors and $(\mathbf{H}\mathbf{w}_p^{(i)})_r$ and $(\mathbf{w}_p^{(i)})_r$ denote the rth element of these two vectors, respectively.

According to the previous discussion, the similarity transformation given in eq. (3) must be written as $\mathbf{H}\mathbf{Z}^{[n]} = \mathbf{Z}^{[n]}\mathbf{H}'$.

Finally, we need an expression to compute the $\mathbf{w}_p^{(i)}$ vector at the iteration i. For this aim we use the perturbation theory applied on the above similarity transformation equation $\mathbf{H}\mathbf{Z}^{[n]} = \mathbf{Z}^{[n]}\mathbf{H}'$. As $\mathbf{Z}^{[n]} = \mathbf{I} + \varepsilon \mathbf{W}^{(1)} + \varepsilon^2 \mathbf{W}^{(2)} + \cdots$ [see eq. (7)], we can write

$$\begin{aligned} (\mathbf{H}_D + \varepsilon \mathbf{H}_{ND})(\mathbf{I} + \varepsilon \mathbf{W}^{(1)} + \varepsilon^2 \mathbf{W}^{(2)} + \cdots) \\ = (\mathbf{I} + \varepsilon \mathbf{W}^{(1)} + \varepsilon^2 \mathbf{W}^{(2)} + \cdots) \\ \times (\mathbf{H}_D + \varepsilon \mathbf{H}_D^{(1)} + \varepsilon^2 \mathbf{H}_D^{(2)} + \cdots), \end{aligned} \quad (10)$$

where $\mathbf{H}_D$ and $\mathbf{H}_{ND}$ are the diagonal and nondiagonal part of the **H** matrix, respectively, and ε is the perturbation parameter. As in the standard perturbation theory, first we equate eq. (10) at different orders of ε and we impose the conditions that the nonzero column of the $\{\mathbf{W}^{(k)}\}$ matrices is placed in the same position, say p, and that $(\mathbf{W}^{(k)})_{pp} = (\mathbf{w}_p^{(k)})_p = 0$. Finally, each p column of each $\mathbf{H}_D^{(k)}$ matrix has zero elements except the

$(\mathbf{H}_{\mathrm{D}}^{(k)})_{pp}$ one. Under these considerations, to first order, we obtain

$$\mathbf{H}_{\mathrm{D}}\mathbf{W}^{(1)} + \mathbf{H}_{\mathrm{ND}} = \mathbf{W}^{(1)}\mathbf{H}_{\mathrm{D}} + \mathbf{H}_{\mathrm{D}}^{(1)}. \quad (11)$$

Equating column p of each side of eq. (11) we find

$$(\mathbf{W}^{(1)})_{jp} = \left(\mathbf{w}_p^{(1)}\right)_j = \frac{(\mathbf{H}_{\mathrm{ND}})_{jp}}{(\mathbf{H}_{\mathrm{D}})_{pp} - (\mathbf{H}_{\mathrm{D}})_{jj}} = \frac{\mathbf{H}_{jp}}{\mathbf{H}_{pp} - \mathbf{H}_{jj}} \quad \forall j \neq p \quad (12)$$

and $(\mathbf{H}_{\mathrm{D}}^{(1)})_{pp} = 0$. Note also that $(\mathbf{H}_{\mathrm{ND}})_{ij} = (\mathbf{H}_{\mathrm{D}}^{(1)})_{ij}$ for all $j \neq p$. In this work, eq. (12) was applied. Another choice consists of using second-order expressions that are

$$\mathbf{H}_{\mathrm{D}}\mathbf{W}^{(2)} + \mathbf{H}_{\mathrm{ND}}\mathbf{W}^{(1)} = \mathbf{H}_{\mathrm{D}}^{(2)} + \mathbf{W}^{(1)}\mathbf{H}_{\mathrm{D}}^{(1)} + \mathbf{W}^{(2)}\mathbf{H}_{\mathrm{D}}. \quad (13)$$

Again equating column p of each side of eq. (13), using eq. (12), and making straightforward algebra we find

$$\begin{aligned}(\mathbf{W}^{(2)})_{jp} &= \left(\mathbf{w}_p^{(2)}\right)_j \\ &= \frac{1}{(\mathbf{H}_{\mathrm{D}})_{pp} - (\mathbf{H}_{\mathrm{D}})_{jj}} \sum_{i\neq p} (\mathbf{H}_{\mathrm{ND}})_{ji}\left(\mathbf{w}_p^{(1)}\right)_i \\ &= \frac{1}{(\mathbf{H}_{\mathrm{D}})_{pp} - (\mathbf{H}_{\mathrm{D}})_{jj}} \sum_{i\neq p} \frac{(\mathbf{H}_{\mathrm{ND}})_{ji}(\mathbf{H}_{\mathrm{ND}})_{ip}}{(\mathbf{H}_{\mathrm{D}})_{pp} - (\mathbf{H}_{\mathrm{D}})_{ii}} \\ &= \frac{1}{\mathbf{H}_{pp} - \mathbf{H}_{jj}} \sum_{\substack{i\neq p \\ i\neq j}} \frac{\mathbf{H}_{ji}\mathbf{H}_{ip}}{\mathbf{H}_{pp} - \mathbf{H}_{ii}}, \end{aligned} \quad (14)$$

and

$$(\mathbf{H}_{\mathrm{D}}^{(2)})_{pp} = \sum_{i\neq p} (\mathbf{H}_{\mathrm{ND}})_{pi}\left(\mathbf{w}_p^{(1)}\right)_i = \sum_{i\neq p} \frac{(\mathbf{H}_{pi})^2}{\mathbf{H}_{pp} - \mathbf{H}_{ii}}. \quad (15)$$

The full correction to second order in the column vector $\mathbf{w}_p^{(2)}$ is obtained by adding eq. (14) to eq. (12),

$$\left(\mathbf{w}_p^{(2)}\right)_j = \frac{\mathbf{H}_{jp}}{\mathbf{H}_{pp} - \mathbf{H}_{jj}} + \frac{1}{\mathbf{H}_{pp} - \mathbf{H}_{jj}} \times \sum_{\substack{i\neq p \\ i\neq j}} \frac{\mathbf{H}_{ji}\mathbf{H}_{ip}}{\mathbf{H}_{pp} - \mathbf{H}_{ii}} \quad \forall j \neq p. \quad (16)$$

Because we are interested in large matrices, eq. (16) can be written in this way:

$$\mathbf{w}_p^{(2)} \leftarrow \mathbf{w}_p^{(1)} + (\mathbf{H}_{pp}\mathbf{I} - \mathbf{H}_{\mathrm{D}})^{-1}\left[\mathbf{H}\mathbf{w}_p^{(1)} - \mathbf{H}_{\mathrm{D}}\mathbf{w}_p^{(1)}\right]. \quad (17)$$

Finally, the $\mathbf{z}_p^{(2)}$ vector is obtained using eq. (5): $\mathbf{z}_p^{(2)} = \mathbf{e}_p + \mathbf{w}_p^{(2)}$. The $\mathbf{z}_p^{(2)}$ vectors could be used as basis vectors of the iterative subspace. In the present method, eq. (12) will give us the initial eigenvector approximation. In the successive iterations the same formula will be applied to the transformed matrix of eq. (9).[21]

VARIANCE MINIMIZATION IN ITERATIVE SUBSPACE

Let us assume that one is interested in finding the eigenpair $E_p, \mathbf{C}_p$. In the proposed algorithm, eqs. (1) or (2) could be projected and solved in the subspace defined by the set of the $\{\mathbf{z}_p^{(k)}\}_{k=1}^{\text{iterations}}$ vectors as in the normal Ritz–Galerkin based methods.[2,6,9] The eigenvalues in the subspace will converge monotonically and are the upper bounds to the exact eigenvalues.[22] However, rather than using this procedure, we compute the variance matrix $\mathbf{V}\{E_p^{(k)}\} = [\mathbf{H}\mathbf{Z}_p^{(k)} - E_p^{(k)}\mathbf{Z}_p^{(k)}]^T[\mathbf{H}\mathbf{Z}_p^{(k)} - E_p^{(k)}\mathbf{Z}_p^{(k)}]$ where $\mathbf{Z}_p^{(k)} = (\mathbf{z}_p^{(1)}|\cdots|\mathbf{z}_p^{(k)})$ and $E_p^{(k)} = (\mathbf{H}\mathbf{z}_p^{(k)})_p$ and, following the idea of Feler,[17] at any iteration, say k, we

$$\text{minimize} \quad \mathbf{v}^T\left(\mathbf{V}\left\{E_p^{(k)}\right\}\right)\mathbf{v} \quad \text{subject to}$$

$$\text{the restriction} \quad \mathbf{v}^T\left(\mathbf{Z}_p^{(k)}\right)^T\mathbf{Z}_p^{(k)}\mathbf{v} = 1. \quad (18)$$

The solution of problem (18) is a generalized eigenvalue problem, but it is solved in the reduced dimension of the current subspace,

$$\mathbf{V}\left\{E_p^{(k)}\right\}\mathbf{v}_i^{(k)} = \mu_i^{(k)}\mathbf{M}_p^{(k)}\mathbf{v}_i^{(k)} \quad i = 1, k \quad (19)$$

where $\mathbf{V}\{E_p^{(k)}\}$ is nonnegative defined and $\mathbf{M}_p^{(k)} = (\mathbf{Z}_p^{(k)})^T\mathbf{Z}_p^{(k)}$. The resulting reduced generalized eigenvalue problem [eq. (19)] is solved, for instance, by the well-known Jacobi–Cholesky method, and a set of eigenpairs $\{\mu_i^{(k)}, \mathbf{v}_i^{(k)}\}_{i=1}^{k}$ is generated. The eigenvalues are in increasing order $\mu_1^{(k)} \leq \mu_2^{(k)} \leq \cdots \leq \mu_k^{(k)}$. At convergence, the lowest eigenvalue $\mu_1^{(k)} = 0$ and the corresponding pair $E_p^{(k)}, \mathbf{Z}_p^{(k)}\mathbf{v}_1^{(k)}$ is an eigenpair of the $\mathbf{H}$ matrix, that is $E_p^{(k)} = E_p$ and $\mathbf{Z}_p^{(k)}\mathbf{v}_1^{(k)} = \mathbf{C}_p$. In other situations $\sqrt{\mu_1^{(k)}} > |E_p^{(k)} - E_p|$, where E_p is the eigenvalue closest to $E_p^{(k)}$ and $\mathbf{Z}_p^{(k)}\mathbf{v}_1^{(k)}$ is the vector closest to the corresponding $\mathbf{C}_p$ eigenvector.[23] Due to the latter result one can apply this method of variance

minimization for locating an internal eigenpair. The procedure can be seen as an extension of the DIIS technique[19,20] or also as a ''least square.''[9]

TREATMENT OF ROOT CLUSTERS BY VARIANCE MINIMIZATION

In this subsection we describe how to find a set of roots simultaneously using the preceding methodology. The basic idea is quite close to that proposed by Kosugi.[24] With the APT method we generate at each iteration a set

$$\left\{\mathbf{z}_p^{(k)}\right\}_{p=1}^{\text{number of desired roots}}$$

that will increase the basis of the iterative subspace and where each $\mathbf{z}_p^{(k)}$ vector corresponds to a different root. Without a loss of generality we now assume that the pth column of the matrix $\mathbf{H}$ is associated to the pth wanted eigenvalue of the cluster. The $\mathbf{Z}^{(k)}$ matrix now takes the following structure:

$$\begin{aligned}\mathbf{Z}_{p,q,\ldots,s}^{(k)} &= \left(\mathbf{Z}_p^{(k)}|\mathbf{Z}_q^{(k)}|\cdots|\mathbf{Z}_s^{(k)}\right)\\ &= \left(\mathbf{z}_p^{(1)}|\cdots|\mathbf{z}_p^{(k)}|\mathbf{z}_q^{(1)}|\cdots|\mathbf{z}_q^{(k)}|\cdots|\mathbf{z}_s^{(1)}|\cdots|\mathbf{z}_s^{(k)}\right),\end{aligned}$$

where the indices $p, q, \ldots, s$ are attached to the respective eigenpairs. The corresponding variance matrices are

$$\mathbf{V}\{E_r^{(k)}\} = \begin{pmatrix} \left[\mathbf{HZ}_p^{(k)} - E_r^{(k)}\mathbf{Z}_p^{(k)}\right]^T\left[\mathbf{HZ}_p^{(k)} - E_r^{(k)}\mathbf{Z}_p^{(k)}\right] & \cdots & \left[\mathbf{HZ}_p^{(k)} - E_r^{(k)}\mathbf{Z}_p^{(k)}\right]^T\left[\mathbf{HZ}_s^{(k)} - E_r^{(k)}\mathbf{Z}_s^{(k)}\right] \\ \vdots & \ddots & \vdots \\ \left[\mathbf{HZ}_s^{(k)} - E_r^{(k)}\mathbf{Z}_s^{(k)}\right]^T\left[\mathbf{HZ}_p^{(k)} - E_r^{(k)}\mathbf{Z}_p^{(k)}\right] & \cdots & \left[\mathbf{HZ}_s^{(k)} - E_r^{(k)}\mathbf{Z}_s^{(k)}\right]^T\left[\mathbf{HZ}_s^{(k)} - E_r^{(k)}\mathbf{Z}_s^{(k)}\right] \end{pmatrix} \quad \forall r \in p, q, \ldots, s, \quad (20)$$

and the $\mathbf{M}_{p,q,\ldots,s}^{(k)}$ matrix is

$$\mathbf{M}_{p,q,\ldots,s}^{(k)} = \begin{pmatrix} \left[\mathbf{Z}_p^{(k)}\right]^T\mathbf{Z}_p^{(k)} & \cdots & \left[\mathbf{Z}_p^{(k)}\right]^T\mathbf{Z}_s^{(k)} \\ \vdots & \ddots & \vdots \\ \left[\mathbf{Z}_s^{(k)}\right]^T\mathbf{Z}_p^{(k)} & \cdots & \left[\mathbf{Z}_s^{(k)}\right]^T\mathbf{Z}_s^{(k)} \end{pmatrix}. \quad (21)$$

We solve eq. (19) by using the matrices defined in eqs. (20) and (21) and selecting the lowest eigenpair $\{\mu_1^{(k)}, \mathbf{v}_1^{(k)}\}$. This procedure is repeated as many times as the number of eigenvalues of the $\mathbf{H}$ matrix that are desired. The reason is that the $\mathbf{V}\{E_p^{(k)}\}$ matrix specifically depends on a particular eigenvalue [see eq. (20)].

RESTART PROCEDURE AND FIXED DIMENSION ITERATIVE SUBSPACE

One of the problems arising in the diagonalization of large matrices is that the storage of the vectors $\{\mathbf{z}_p^{(k)}\}_{k=1}^{\text{iterations}}$ set and residues $\{\mathbf{Hz}_p^{(k)}\}_{k=1}^{\text{iterations}}$ becomes a bottleneck and the algorithm must be restarted very often or the earlier expansion of the basis set of the vectors must be discarded. In general, both procedures slow down the convergence.[12] However, no systematic study of this effect has been carried out yet.

In connection with this problem, we present an algorithm related to the fixed dimension iterative subspace. This algorithm seems very promising because along the whole process it never increases the memory requirements and, consequently, it does not present any restart problem. The bases of the algorithm are the ideas proposed by Karush.[4] In our version, we only need an iterative subspace of dimension K. The number K is always much smaller than the full dimension of the problem to be diagonalized. Starting with the $\mathbf{z}_p^{(1)}$ vector in the first K iterations of the process, the iterative subspace increases as proposed in the previous subsections. At the iteration $K+1$ the new $\mathbf{z}_p^{(K+1)}$ vector replaces the oldest one; that is, the $\mathbf{z}_p^{(1)}$ vector; at the iteration $K+2$ the vector $\mathbf{z}_p^{(K+2)}$ replaces the vector $\mathbf{z}_p^{(2)}$; and so on. This process is continued until one achieves convergence. Note that from the iteration K the iterative subspace never increases. The variance minimization is applied to obtain the new vector $\mathbf{z}_p^{(k)}$. The number K is normally selected according to memory requirements. In this way one avoids the need of restarting. Note that this procedure can be seen as a

generalization of the three term recursion of van Lenthe and Pulay.[11,12]

Computational Details: Single Root and Cluster Diagonalization Algorithm

In this section we show an algorithm based on the previous theoretical results that is useful for searching for one or more eigenpairs. It also covers a restart option. A sketch of the algorithm dealing with first-order corrections [see eq. (12)] follows:

1. Specify the cluster members.

2. If variance treatment will be used then

Set $K > 1$, the maximal subspace dimension allowed for every cluster member.

Specify if a fixed subspace dimension will be maintained along the computation.

else

Set $K = 1$

end if

3. Loop over every cluster member i.

Define a starting vector from eq. (12): $\mathbf{z}_i^{(1)} = \mathbf{e}_i + \mathbf{w}_i^{(1)}$.

Set the subspace dimension counter for the cluster member: $k_i = 1$.

end loop

4. Iterative loop

Loop over i: every nonconverged cluster member

If variance method is requested

Construct the metric matrix **M** according eq. (21).

Check for linear dependencies. If they are found, reduce the dimensionality via metric diagonalization. Then, metric **M** is also formally redefined to be the unit one.

From E_i, the most recent eigenvalue approximation for the cluster member, obtain the variance matrix **V** according to eq. (20) and using the linear independent vectors.

Diagonalize the variance matrix **V** over the metric **M** [eq. (19)].

Construct the new eigenvector approximation, $\mathbf{z}_{\text{new}}$, from the eigenvector related to the lowest eigenvalue of **V**.

else

Set $\mathbf{z}_{\text{new}} = \mathbf{z}_i^{(k_i)}$.

end if

Send the last eigenvector approximation to the APT method. Using eqs. (9) and (12) an improved eigenvector version will be obtained: $\mathbf{z}_{\text{new}} \leftarrow \mathbf{z}_{\text{new}} + \mathbf{w}$.

If $(k_i = K)$ then

If variance option is active and a fixed subspace dimension must be kept then

Shift basis set vectors: $\mathbf{z}_i^{(j)} = \mathbf{z}_i^{(j+1)}$ $\forall j = 1, K-1$.

else

Set $k_i = 1$: restart is performed.

end if

else

Set $k_i = k_i + 1$: Subspace dimension increases.

end if

Keep the new vector into a new reservoir: $\mathbf{z}_i^{(k_i)} = \mathbf{z}_{\text{new}}$.

Check if convergence is achieved for this cluster member.

end loop

If all cluster members are converged go to 5.

end of iterative loop

5. Write results

6. End

Numerical Examples

Several numerical examples are discussed in this section. Two basic types of matrices were studied. Their first eigenpairs were treated in several ways. In some cases each root was found independently and in other cases they were worked out together at the same time. In the algorithm two options were combined: using the pure APT technique and mixing it with the variance methodology. Here this last option was explored numerically working in a fixed subspace or performing a proper restart. In any case, one eigenpair or a cluster of them were found.

As a first numerical test, several eigenpairs of preconditioned negative Hilbert matrices were found using the above-described methodology.

Such matrices are defined in the following way:

$$\mathbf{H}_{ij}(\gamma) = \begin{cases} \dfrac{-1}{2i-1} & \text{if } i = j, \\ \dfrac{-1}{\gamma(i+j-1)} & \text{if } i \neq j. \end{cases} \quad (22)$$

The off-diagonal elements were divided by a positive parameter γ to make the matrix diagonal dominant enough (10.0 in the actual computations). The tested matrix orders are 10^3, 10^4, and 10^5. Tables I and II report the obtained results. In Table I the restart option was explored while in Table II a fixed subspace of dimension 3 was maintained. For each case the total number of needed iterations to achieve convergence are reported, together with the norm of the final error vector, $\mathbf{H}\mathbf{z}_p^{(k)} - E_p^{(k)}\mathbf{z}_p^{(k)}$, and the greatest module found along the components of such a vector. The convergence criteria used to terminate every computation was to force this largest module to be less than a prespecified parameter ε. For matrices of order 10^3 and 10^4, the convergence criterion ε was set to 10^{-8}; for the largest matrix it was set to $\varepsilon = 10^{-7}$. In all cases the prevention and treatment of linear dependencies was performed.

As a second example, the first eigenpairs of the 50×50 Raffenetti matrix[25] were found. It is im-

TABLE I.
Number of Iterations Needed for Calculation of Several Eigenpairs of Scaled Negative Hilbert Matrices [eq. (22)].

Matrix Order	Column	One Vector / Cluster		All Three Vectors / Cluster	
1000	1	7	$(2.09 \cdot 10^{-10}, 9.70 \cdot 10^{-11})$	3	$(6.46 \cdot 10^{-11}, 4.05 \cdot 10^{-11})$
	2	11	$(4.42 \cdot 10^{-9}, 1.85 \cdot 10^{-9})$	6	$(6.32 \cdot 10^{-10}, 2.98 \cdot 10^{-10})$
	3	16	$(1.64 \cdot 10^{-8}, 3.38 \cdot 10^{-9})$	9	$(1.19 \cdot 10^{-8}, 3.01 \cdot 10^{-9})$
10,000	1	6	$(9.84 \cdot 10^{-9}, 2.30 \cdot 10^{-9})$	4	$(4.74 \cdot 10^{-12}, 2.00 \cdot 10^{-12})$
	2	12	$(1.92 \cdot 10^{-9}, 2.85 \cdot 10^{-10})$	7	$(2.29 \cdot 10^{-9}, 7.16 \cdot 10^{-10})$
	3	24	$(2.58 \cdot 10^{-8}, 5.33 \cdot 10^{-9})$	7[a]	
100,000	1	7	$(1.75 \cdot 10^{-8}, 2.82 \cdot 10^{-9})$	3	$(1.75 \cdot 10^{-9}, 6.29 \cdot 10^{-10})$
	2	11	$(2.66 \cdot 10^{-7}, 4.47 \cdot 10^{-8})$	7	$(2.70 \cdot 10^{-8}, 3.50 \cdot 10^{-9})$
	3	40	$(1.67 \cdot 10^{-7}, 4.34 \cdot 10^{-8})$	18	$(2.50 \cdot 10^{-7}, 3.69 \cdot 10^{-8})$

[a]Converges to the second root.
Off-diagonal factor γ was set to 10.0. The terms in parentheses are the norm of the residual vector, $\mathbf{H}\mathbf{z}_p^{(k)} - E_p^{(k)}\mathbf{z}_p^{(k)}$, and the maximal absolute value found for a component of such a vector. Convergence criterion ε is 10^{-8} for the two small matrices and 10^{-7} for the largest one. The maximal subspace dimension is 3 for every cluster member. When this dimension is achieved, the restart option is applied and only one vector is kept; then the subspace is rebuilt.

TABLE II.
Number of Iterations Needed for Calculation of Several Eigenpairs of Scaled Negative Hilbert Matrices [eq. (22)].

Matrix Order	Column	One Vector / Cluster		All Three Vectors / Cluster	
1000	1	5	$(2.38 \cdot 10^{-9}, 9.71 \cdot 10^{-10})$	3	$(6.46 \cdot 10^{-11}, 4.05 \cdot 10^{-11})$
	2	7	$(9.16 \cdot 10^{-10}, 4.49 \cdot 10^{-10})$	5	$(1.39 \cdot 10^{-9}, 4.86 \cdot 10^{-10})$
	3	13	$(8.39 \cdot 10^{-9}, 2.23 \cdot 10^{-9})$	6	$(8.49 \cdot 10^{-9}, 1.47 \cdot 10^{-9})$
10,000	1	21	$(1.90 \cdot 10^{-9}, 4.38 \cdot 10^{-10})$	4	$(8.02 \cdot 10^{-11}, 3.70 \cdot 10^{-11})$
	2	11	$(2.46 \cdot 10^{-9}, 3.79 \cdot 10^{-10})$	7	$(2.32 \cdot 10^{-9}, 9.75 \cdot 10^{-10})$
	3	24	$(6.02 \cdot 10^{-9}, 5.30 \cdot 10^{-9})$	10[a]	
100,000	1	7	$(9.45 \cdot 10^{-9}, 1.24 \cdot 10^{-9})$	3	$(1.75 \cdot 10^{-9}, 6.30 \cdot 10^{-10})$
	2	9	$(1.65 \cdot 10^{-7}, 1.07 \cdot 10^{-8})$	5	$(1.05 \cdot 10^{-8}, 4.26 \cdot 10^{-9})$
	3	21	$(3.32 \cdot 10^{-7}, 1.01 \cdot 10^{-8})$	12	$(1.36 \cdot 10^{-7}, 3.51 \cdot 10^{-8})$

[a]Converges to the second root.
Off-diagonal factor γ was set to 10.0. The terms in parentheses are the norm of the residual vector, $\mathbf{H}\mathbf{z}_p^{(k)} - E_p^{(k)}\mathbf{z}_p^{(k)}$, and the maximal absolute value found for a component of such a vector. Convergence criterion ε is 10^{-8} for the two small matrices and 10^{-7} for the largest one. The maximal subspace dimension is 3 for every cluster member. Once this dimension is achieved, it is maintained (fixed subspace option).

TABLE III.
Number of Iterations Needed for Calculation of First Eigenpairs of 50 × 50 Raffenetti Matrix.[25]

Column	Eigenvalue	One Vector / Cluster		Two Vectors / Cluster	
1	0.0336080	20	$(4.07 \cdot 10^{-8})$	18	$(7.26 \cdot 10^{-8})$
2	0.1432514936	—		9	$(1.88 \cdot 10^{-9})$
3	0.2519748283	57	$(1.04 \cdot 10^{-6})$		
4	0.3623426694	27	$(3.76 \cdot 10^{-7})$		
5	2.349421193	22	$(1.12 \cdot 10^{-8})$		
6	10.34995782	9	$(1.55 \cdot 10^{-8})$		

The term in parentheses is the norm of the residual vector $\mathbf{Hz}_p^{(k)} - E_p^{(k)}\mathbf{z}_p^{(k)}$. Convergence criterion ε is 10^{-8}. The maximal subspace dimension is 9 for every cluster member. At every restart, only one vector is kept. The last column shows how the second root was found when it was forming a cluster with the first eigenpair.

portant to emphasize that, despite its small order, the Raffenetti matrix is very difficult to diagonalize. As can be seen from the results of Table III, the use of the APT method plus the variance technique was enough to obtain all the eigenpairs except one. It was possible to find the second one when it was treated together with the first eigenpair, forming a cluster of two members and using the same methodology.

From the examples, two important characteristics can be stressed with respect to the use of the variance treatment methodology: first, this technique allows the direct computation of internal eigenpairs, because it was done when obtaining the third column of data in Tables I and II; and, second, it also decreases the amount of computational effort to be done when searching two or more eigenvalues at the same time (see the fourth column of both Tables and compare with the third one). The use of the APT technique alone without the help of a variance treatment is meaningless: the number of needed iterations increases notably and in some cases the convergence could not be achieved.

The behavior of the present algorithm was compared with the Davidson's one. The first or the three first matrix eigenpairs were computed. The convergence criteria and conditions were the same as the ones used to elaborate Tables I and II. The number of necessary iterations to achieve convergence were the same for both methods. Thus, both methodologies exhibit the same performance features when we are looking for the first matrix eigenpairs.

As is typical in this kind of computation, the evaluation of $\mathbf{Hz}_p^{(k)}$ vectors is the bottleneck of the full calculation. For all the computations involving the largest matrices it took more than 99.9% of the whole computation time. This shows that the time spent by input–output flow or the subspace treatment and its related diagonalization tasks are negligible.

In all the examples the variance matrix was diagonalized and the $\mathbf{Hz}_p^{(k)}$ vector attached to the new generated subspace vector was recomputed without the use of the linear combination that generated the related $\mathbf{z}_p^{(k)}$ vector. This was done to preserve the precision, despite it being more time demanding. This option can be useful when treating systems requiring a considerable number of iterations.

Conclusions

A one-vector algorithm for the search of matrix eigenpairs was presented and its theoretical basis analyzed. Synergistic properties were found when combining it with the variance minimization technique. The whole procedure was studied in consonance with the subspace treatment features that automatically emerge from this new approach. Restart and fixed space techniques were explored. As a result, not only were the convergence properties of the original method ameliorated, but also how arbitrary eigenvectors and the related eigenvalues can be directly obtained were shown. It was also shown how the full procedure allows the simultaneous computation of several eigenpairs, ameliorating the convergence rates with respect to the individual eigenpairs treatment. Numerical results were provided and analyzed. These illustrated the efficacy of the numerical and algebraic machinery that the authors are proposing.

Acknowledgments

We are indebted to Professor S. Olivella for his valuable suggestions. We acknowledge an unknown referee for giving us the correction of eq. (9) and valuable comments.

References

1. I. Shavitt, In *Methods of Electronic Structure Theory*, H. F. Schaefer III, Ed., Plenum, New York, 1977, p. 189.
2. (a) W. Ritz, *J. Rein. Angew. Math.*, **135**, 1 (1909); (b) B. G. Galerkin, *Vestnik Ing.*, **1**, 897 (1915).
3. C. Lanczos, *J. Res. Natl. Bur. Stand.*, **45**, 255 (1950), **49**, 33 (1952).
4. W. Karush, *Pacif. J. Math.*, **1**, 233 (1951).
5. C. C. Paige, *J. Inst. Math. Applic.*, **18**, 341 (1976).
6. B. N. Parlett, *The Symmetric Eigenvalue Problem*, Prentice–Hall, Englewood Cliffs, NJ, 1980.
7. (a) E. R. Davidson, *J. Comput. Phys.*, **17**, 87 (1975); (b) E. R. Davidson, *J. Phys. A*, **13**, L179 (1980); (c) E. R. Davidson, *Comput. Phys. Commun.*, **53**, 49 (1989); (d) C. W. Murray, S. C. Racine, and E. R. Davidson, *J. Comput. Phys.*, **103**, 382 (1992).
8. B. Liu, Report on the Workshop ''Numerical Algorithms in Chemistry: Algebraic Methods,'' of the National Resource for Computation in Chemistry, Lawrence Berkeley Laboratory, 9–11 August, C. Moler and I. Shavitt, Eds., LBL-8158, UC-32, CONF-780878, Lawrence Berkeley Laboratory, Livermore, CA, 1978, p. 49.
9. Y. Saad, *Numerical Methods for Large Eigenvalue Problems*, Manchester University Press, Halsted Press, Wiley, New York, 1992.
10. J. Olsen, P. Jørgensen, and J. Simons, *Chem. Phys. Lett.*, **169**, 463 (1990).
11. J. M. Bofill and J. M. Anglada, *Chem. Phys.*, **183**, 19 (1994).
12. J. H. van Lenthe and P. Pulay, *J. Comput. Chem.*, **11**, 1164 (1990).
13. A. V. Mitin, *J. Comput. Chem.*, **15**, 747 (1994).
14. (a) M. R. Hestenes and E. Stiefel, *J. Res. Natl. Bur. Stand.*, **49**, 498 (1952); (b) M. R. Hestenes, *Conjugate Direction Methods in Optimization*, Springer, New York, 1980.
15. G. A. Gallup, *J. Comput. Chem.*, **3**, 127 (1982).
16. L. M. Cheung and D. M. Bishop, *Comput. Phys. Commun.*, **13**, 247 (1977).
17. M. G. Feler, *J. Comput. Phys.*, **14**, 341 (1974).
18. I. Shavitt, C. F. Bender, A. Pipano, and R. P. Hosteny, *J. Comput. Phys.*, **11**, 90 (1973).
19. D. M. Wood and A. Zunger, *J. Phys. A Math. Gen.*, **18**, 1343 (1985).
20. (a) P. Pulay, *Chem. Phys. Lett.*, **73**, 393 (1980); (b) P. Pulay, *J. Comput. Chem.*, **3**, 556 (1982).
21. E. Besalú and R. Carbó–Dorca, *J. Math. Chem.*, **21**, 395 (1997).
22. J. Linderberg and Y. Ohrn, *Int. J. Quantum Chem.*, **12**, 161 (1977).
23. D. H. Weinstein, *Proc. Natl. Acad. Sci.*, **20**, 529 (1934).
24. N. Kosugi, *J. Comput. Phys.*, **55**, 426 (1984).
25. R. C. Raffenetti, *J. Comput. Phys.*, **32**, 403 (1979).